

REMARKS

This application has been carefully reviewed in light of the Office Action dated September 11, 2006. Claims 3 to 6, 8 to 16, 19, 20, 22 to 25 and 27 are in the application, with Claims 17 and 18 having been cancelled herein. Claims 3 and 22 are independent. Reconsideration and further examination are respectfully requested.

Claims 3 to 6, 14 to 16 and 22 to 27 were rejected under 35 U.S.C. § 103(a) over U.S. Patent No. 6,199,204 (Donohue) in view of U.S. Patent No. 5,711,672 (Redford) and U.S. Patent No. 6,405,362 (Shih), and Claims 8 to 11 and 17 to 20 were rejected under § 103(a) over Donohue in view of Redford and Shih and further in view of U.S. Patent No. 6,121,967 (Foster). Reconsideration and withdrawal of the rejections are respectfully requested.

The present invention relates to controlling the downloading and use of software via a removable storage medium. According to the invention, identification information identifying software to be downloaded is stored on the removable storage medium. In this regard, the removable storage medium has a low storage capacity such that the software can't be stored on it. When the removable storage medium is connected, the information identifying the software to be downloaded is read and transmitted to a server. If the software is not already stored in an internal storage medium, the software is downloaded, and so long as the removable storage medium remains connected, the software application can be used. If the application closes, a menu screen is presented to the user requesting that the user restart the software application. However, if the removable storage medium is disconnected, the menu screen is deleted so that the user cannot use the software. Additionally, if the software is stored in the internal storage medium, when the removable storage medium is connected, version information is sent to the server, and if the version of the software stored in the internal storage medium and that stored in the server are identical, the application is started. However, if the versions are not

identical, the server downloads a newer version to the internal storage medium and starts the application. As a result, the software application can only be downloaded and used so long as the removable storage medium remains connected.

Referring specifically to the claims, amended independent Claim 3 is directed to an information processing device comprising a communication unit that communicates with a server via a network, a portable-information-storage-medium connection unit to which a portable information storage medium is connectable, wherein the portable information storage medium stores information identifying software to be acquired via the network from the server, and has a low capacity for storing the information without storing the software, a detecting unit that detects whether the portable information storage medium is connected with the portable-information-storage-medium connection unit, a reading unit that reads, from the portable information storage medium, the information identifying the software to be acquired when the connection of the portable information storage medium with the portable-information-storage-medium connection unit is detected by the detecting unit, a sending unit that sends the information identifying the software to be acquired read by the reading unit to the server through the communication unit, an information transfer unit that downloads, from the server, the software identified by the information identifying the software to be acquired sent by the sending unit into an internal storage medium of the information processing device, a software storage unit that stores the downloaded software into a software storage area of the internal storage medium, a software management unit that manages the software stored in the software storage area of the internal storage medium, a menu display unit that displays a menu screen to instruct restarting of the software when the software stored in the software storage area closes after starting up, an unloading detecting unit that detects an unload of the portable information storage medium from the portable-information-storage-medium connection unit, a deleting unit that deletes the menu screen displayed by the

menu display unit in a condition where the software is stored into the software storage area when the unload is detected by the unloading detecting unit, an estimation unit that estimates whether the software identified by the information identifying the software to be acquired read by the reading unit is already stored into the software storage area of the internal storage medium, and a sending unit that sends version information of the software stored in the internal storage medium to the server when it is estimated by the estimation unit that the software is already stored into the software storage area, wherein the software management unit starts the software stored into the software storage area when a version of the software in the server and the version of the software stored into the software storage area are identical, and the software management unit downloads a new software version from the server via the information transfer unit and starts the software based on the version information transferred by the information transfer unit.

The applied art, alone or in any permissible combination, is not seen to disclose or to suggest the features of the invention, and in particular, is not seen to disclose or to suggest at least the features of 1) deleting a menu screen displayed by a menu display unit in a condition where software to be acquired is stored into a software storage area of an internal storage medium when an unload of a portable information storage medium from a portable-information-storage-medium connection unit is detected, and 2) estimating whether software identified by information identifying software to be acquired read from the portable information storage medium is already stored into the software storage area of the internal storage medium, and a) starting the software stored into the software storage area when a version of the software in the server and the version of the software stored into the software storage area are identical, and b) the downloading a new software version from the server and starting the software based on the version information transferred by the information transfer unit.

Donohue merely relates to distributing software updates. According to the patent, a software updater agent automatically accesses network locations and downloads and installs software updates. However, as rightly conceded in the Office Action, Donohue fails to detect unloading of a portable information storage medium and fails to delete software upon such detection. Additionally, Donohue fails to disclose anything with regard to the claimed feature of displaying a menu screen instructing to restart the software, much less that the menu screen is deleted when an unloading of the portable information storage medium is detected.

Redford merely teaches that an application program is automatically started when a storage medium is inserted, and that the application program is automatically ended when the storage medium is removed. The starting and ending of an application program is simply different from downloading and deleting software upon connection of the medium and unloading of the medium. Specifically, in Redford, the application program is already resident on the computer and remains stored thereon when the medium is removed. Moreover, Applicant fails to see any teaching in Redford regarding the display of a menu screen instructing to restart software if the software is stored in an internal storage medium, and then deleting the menu screen if an unloading of a portable information storage medium is detected. Further, Redford is not seen to teach anything with regard to the claimed feature of comparing the versions of the software and downloading the latest version from a server. Thus, Redford fails to make up for the deficiencies of Donohue.

Shih merely teaches that an autorun program is invoked to install software from a recording medium when the recording medium is inserted, and to end the autorun program when the recording medium is removed. Thus, Shih merely installs software, but like Donohue and Redford, fails to delete a menu screen to instruct restarting of the software if an unload of a portable information storage medium is detected.

Foster is merely seen to teach the display of a status of a media bay. If a user action occurs to remove a device from a media bay that is in use, a control menu may be displayed to prompt the user to reinsert the device. If the user reinserts the device, the displayed menu is cancelled. However, Foster is not seen to disclose or to suggest that a menu screen instructing a user to restart software is displayed if software is stored in an internal storage medium, and then to delete the menu screen when unloading of a removable storage medium is detected. In fact, Foster operates opposite that of the invention since the menu screen of Foster is cancelled when the device is reinserted rather than cancelling the menu screen when the device is removed. Thus, Foster fails to make up for the deficiencies of Donohue, Redford and Shih.

In view of the foregoing amendments and remarks, amended independent Claims 3 and 22, as well as the claims dependent therefrom, are believed to be allowable.

No other matters having been raised, the entire application is believed to be in condition for allowance and such action is respectfully requested at the Examiner's earliest convenience.

Applicant's undersigned attorney may be reached in our Costa Mesa, California office by telephone at (714) 540-8700. All correspondence should continue to be directed to our address given below.

Respectfully submitted,

/Edward Kmett/

Edward A. Kmett
Attorney for Applicant
Registration No. 42,746

FITZPATRICK, CELLA, HARPER & SCINTO
30 Rockefeller Plaza
New York, New York 10112-3801
Facsimile: (212) 218-2200

CA_MAIN 124471v1